Project:	State of Washington Multi-Agency Forest Project
Title:	Change Management
Version:	1.0
Status:	Approved
Date:	July, 2001

Table of Contents

.

Introduction to Change Management

The objective of the change management process is to introduce change into the IT environment quickly and with minimal disruption to service. Change management is responsible for changes in technology, systems, applications, hardware, tools, documentation, and processes, as well as changes in roles and responsibilities. Change management is charged with providing prompt and efficient handling of any **Change request (CR)**.

A key goal of the change management process is to ensure that all parties affected by a given change are aware of and understand the impact of the impending change. Since most systems are heavily interrelated, any change made in one part of a system may have profound impacts on another. Change management attempts to identify all affected systems and processes before the change is implemented in order to mitigate or eliminate any adverse effects.

Change management receives all CR s, logs them into the **change management log**, evaluates CR s, approves or rejects CR s, and schedules approved CR s for implementation. Approved CR s are passed to the applicable groups responsible for procurement and/or development of software and hardware components. Except for standard changes, the authorization to plan and implement the change is given to the **release management team**. Standard changes are given to an appropriate group such as the service desk (standard changes are discussed in more detail later).

Change management monitors the change process from development/procurement through implementation. Change management also ensures that the entire process is properly documented in the change management log. The log contains information about the change, including CR status, schedule information and work orders.

Change Management Functions

The successful implementation of a change in the IT environment depends on the coordination of many activities within the organization. Below are some of the various functions that are necessary.

Availability Management

Availability management is concerned with the availability and reliability of the IT computing environment. The CR assessment process must be coordinated with the availability coordinator to establish the effect of implementing a proposed change on the availability and reliability of the IT environment. Availability management frequently generates CR s.

Capacity Management

Capacity management ensures that appropriate IT resources are available to meet business requirements by planning for additional resources as current system resource use begins to near the point of full capacity. The capacity manager will review CR s to determine the impact of implementing the change on the performance and capacity of the existing IT infrastructure. The **capacity coordinator** will also evaluate the IT infrastructure after implementation is complete to assess the actual impact to the IT environment. Capacity management frequently generates CR s.

Configuration Management

Configuration management is the process of tracking and accounting for hardware, software, documentation, and all other components of the IT environment. Configuration management is responsible for maintaining the CMDB, which is used to track all IT-related components.

The **configuration coordinator** will be responsible for performing a baseline assessment of the IT environment. Change management uses this baseline assessment in evaluating the impacts of a change.

During the change and release processes, configuration management is tasked with updating the CMDB with all IT system component changes and documenting the status of all components.

Network Administration

Network administration is the process of designing and managing all networks within an organization. The **network coordinator** will be responsible for assessing the impact of CR s on all system networks. Network administration frequently generates CR s.

Problem Management

Problem Management determines the root cause of problems. A problem is defined as one or more incidents exhibiting similar symptoms. Problem management is closely aligned with change management because proposed solutions typically become CR s.

Release Management

Release management coordinates and manages all releases to the IT environment. After a change is approved for implementation in the IT environment, release management gets authorization to plan and implement the release.

Security Administration

Security administration develops, implements, and manages security controls including the software controls and technical infrastructure to ensure the safe and secure operation of the IT environment. The **security coordinator** must review all CR s to assess their impact on the IT environment before change management approves them. The security coordinator also will evaluate system security following the implementation of all changes.

Service Desk

The service desk is tasked with providing efficient, timely, and high-quality user support and incident management. The service desk provides feedback from the user community that frequently generates CR s. After obtaining approval from change management, the service desk implements standard changes to the IT environment.

Change Management Overview

The primary goal of change management is to protect the IT environment from potentially disruptive changes. Objectives of change management include:

1. Providing a standardized and comprehensive approach to the implementation of changes to the IT environment.
2. Minimizing disruptions to the IT environment.
3. Handling all CR s in an efficient and quick manner (evaluating and approving CR s).
4. Establishing implementation schedules.
5. Communicating with those affected by the change.
6. Monitoring the activities of planning, developing, testing, and implementing changes to the IT environment.
7. Reviewing and evaluating the change process.

Scope

Change management is responsible for managing the change process. Change management is most closely tied to release management and configuration management. Change management involves assessing the impact of proposed changes to the IT environment, prioritizing and categorizing changes, determining the course of action, and monitoring the planning, development, testing, and implementation of changes.

Utilizing a change management process will increase service availability and IT efficiency by reducing the number of unnecessary changes. Change management should manage changes that:

1. Will affect multiple users.
2. Could potentially disrupt mission-critical functionality.
3. Involve hardware (such as servers) or software modifications.
4. Involve operational and process modifications that affect multiple users.

Change management **should not** handle frequently performed administrative tasks, or minor service requests such as changing a password.

A change is defined as anything—hardware, software, system components, documents, and so on—introduced to the IT environment that may impact the IT **service level agreement (SLA)**. Changes can be either permanent or temporary.

Permanent changes completely replace a current version of a component. Temporary changes are put in place until a permanent change can be implemented or to serve a particular purpose.

Roles and Responsibilities

Any person in the organization should be authorized to submit an CR . Below the manager level, it is recommended that employees submit CR s to their managers for review and approval before passing them to the change manager.

Roles differ slightly from change management functions as roles are generally assumed by different individuals, given the nature of the change request.

The **change initiator** will be responsible for completely filling out the CR form, which includes the reason for the CR , the requested implementation date, and the systems and personnel that will be affected by the change.

The **change manager** is responsible for managing the activities of the change management process. This individual is involved in every step of the process, from receipt of an CR to the implementation of the change in the IT environment, and is ultimately responsible for the successful implementation of any change to the IT environment.

The **change owner** is responsible for planning and implementing a change in the IT environment. The change owner could be the release manager for major and minor changes or, for example, the service desk for standard changes, such as setting up new personal computer configurations.

The State of Washington will also designate the Windows 2000 Steering Committee as the **Change Advisory Board.**
The function of the **Forest Resource Group (FRG)** is to analyze changes on behalf of the Steering Committee Following authorization, the Steering Committee instructs DIS to execute the change. The FRG will consist initially of members from the Windows 2000 DEV Group.

Responsibility Breakdown

Although some roles have groups with permanent members, the people involved in each role will change, depending on the nature of the CR . It will be important for the State of Washington to clearly assign these roles on a case-by-case basis, particularly in relation to the change owner and change manager. Table 2 outlines the responsibilities of the various roles in the change management process.

Roles	Responsibilities	Team Model Cluster
Change Manager	The change manager is responsible for managing the activities of the change management process. This individual is involved in every step of the process, from receipt of an CR to the implementation of the change in the IT environment, and is ultimately responsible for the successful implementation of any change to the IT environment. The change manager: 1. Receives CR s and ensures that they are properly recorded in the change management log 2. Reviews CR s for completeness 3. Prepares Steering Committee meeting agendas and providing all necessary review information to the Steering Committee members prior to board meetings 4. Assigns teams to conduct CR impact analyses and risk assessments 5. Analyzes, prioritizes, categorizes, approves, and schedules CR s 6. Provides change notification to affected parties 7. Monitors the successful completion of all CR s, including change planning, building, testing, and implementation to ensure that these processes follow the change schedule 8. Reviews and evaluates the change process	This role is part of the Release cluster within the MOF Team Model
Change Owner	The change owner is responsible for planning and implementing a change in the IT environment. The change owner could be the release manager for major and minor changes or, for example, the service desk for standard changes, such as setting up new personal computer configurations. The change owner does the following: 1. Receives changes from the change manager or the Steering Committee. 2. Follows the change schedule that the Steering Committee defines. 3. Coordinates the development/procurement, testing, and implementation of the assigned change. 4. Provides project status feedback to the change manager and identify any problems	This role can be anyone from any role cluster within the MOF Team Model.

	as they arise. 5. Works with the change initiator to ensure that the change meets the initiator's requirements. 6. Works with the change initiator to correct any problems or provide the correct system enhancements intended by the CR . 7. Assists the change manager in evaluating the change process after the change is implemented.	
Steering Committee	The Steering Committee does the following: 1. Assesses and approves changes that the change manager is not authorized to approved. 2. Review the status of a change throughout the change process. 3. Assess progress with respect to the approved schedule. 4. Determine how to correct any identified problems. 5. Communicate findings with appropriate managers and stakeholders.	The board should include the change manager as well as a representative from each role cluster within the MOF Team Model.
Forest Resource Group	The Forest Resource Group does the following: 1. Analyze and recommend changes to the Steering Committee.	The committee should include executives from the Infrastructure role cluster within the MOF Team Model.

Table 2 - Roles Summarized

Types of Changes

As changes flow into the process, they will be assigned a priority. The change categorization is determined by the impact to the organization and the resources required to plan, develop, and implement the change. Changes will be grouped into the following categories:

Major Change

A major change is a change that significantly impacts the IT environment or requires a great deal of resources to plan, build, and implement. Major changes include new system functionality and enhancements to existing functionality. The Steering Committee must approve these changes. Following approval, major changes are passed to DIS for scheduling and subsequently to release management for planning and implementation.

A common example of a major change is a schema change.

Significant Change

Significant changes require a great deal of resources to plan, build, and implement in the IT environment. The change manager reviews significant changes and passes them to the Steering Committee for analysis and approval.

A common example of a significant change is a new agency domain being added to the forest or withdrawing from the forest.

Standard Change

A minor change does not require significant resources. This type of change will not significantly impact the IT environment and will typically affect only a small number of users in the environment. Approval authority for minor changes should be delegated to the change manager. This individual will have the option to approve the change or pass it to the Steering Committee for further review and approval.

A common example of a minor change would be a Microsoft Windows Service Pack or Security Update.

Minor Change

A standard change is a change that follows an established and documented approach. This type of change poses little or no threat to the reliable operation of the IT environment. The change manager approves a standard change and passes it to the appropriate change owner (such as the service desk) who is preauthorized to plan and implement the change. An example of a standard change is setting up and configuring a personal computer for a new employee.

Examples of standard changes include replication schedule or topology changes, site links, etc.

Change Management Process

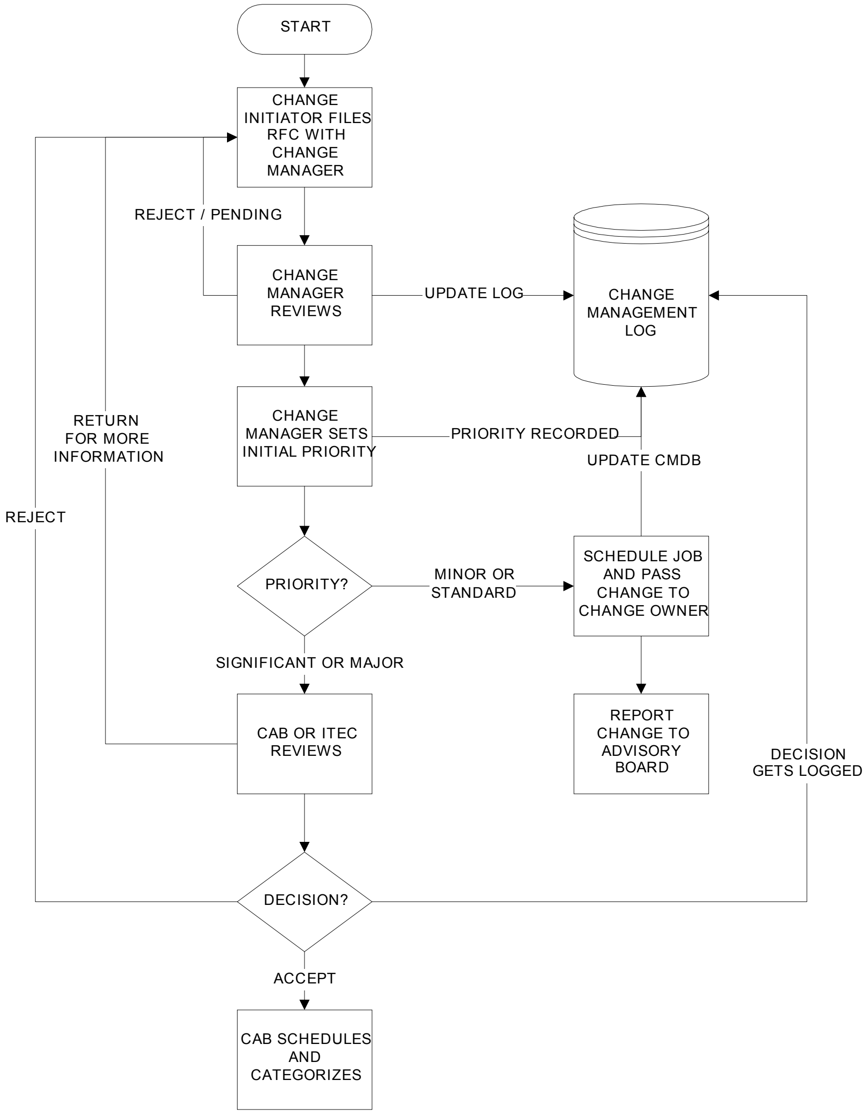
START
CHANGE INITIATOR FILES RFC WITH CHANGE MANAGER
REJECT / PENDING
CHANGE MANAGER REVIEWS
UPDATE LOG
CHANGE MANAGEMENT LOG
CHANGE MANAGER SETS INITIAL PRIORITY
PRIORITY RECORDED
UPDATE CMDB
RETURN FOR MORE INFORMATION
REJECT
PRIORITY?
MINOR OR STANDARD
SCHEDULE JOB AND PASS CHANGE TO CHANGE OWNER
SIGNIFICANT OR MAJOR
CAB OR ITEC REVIEWS
REPORT CHANGE TO ADVISORY BOARD
DECISION GETS LOGGED
DECISION?
ACCEPT
CAB SCHEDULES AND CATEGORIZES

Figure 1 - Change Management Workflow

Change management is broken down into the following seven activities:

1. Change request receipt
2. Change analysis and review by the Steering Committee/Forest Resource Group
3. Change notification and release
4. Change building, testing, and implementation monitoring
5. Change outcome notification
6. Post-implementation evaluation

Each activity is an integral part of the change management process. The following sections will describe each of the seven change management activities. An overview of the process is illustrated in Figure 1.

1. Change request Receipt

When the change manager receives an CR , the change manager logs it into the change management log. The change manager is responsible for recording the status of the CR in the change management log throughout the change process.

The change manager reviews all CR s for completeness and practicality. The change manager returns rejected CR s to the change initiator with an explanation and prioritizes approved CR s.

CR s prioritized as urgent are passed to the appropriate group(s); all other CR s will either be passed to the Forest Resource Group for analysis and approval or categorized by the change manager and passed to the appropriate change owner.

See appendix B for an example of a Change Request. Ultimately, the example in appendix B would be an online web form that facilitated issue tracking and management.

2. Change Analysis and Review

The Steering Committee and Forest Resource Group are tasked with assessing and approving changes that the change manager does not handle. As depicted in Figure 5, these groups will assess the urgency, risk, potential impacts, costs, and resources required to plan, develop, test, and implement the change. Based on this information, the CR will be approved or rejected.

3. Change Notification

The change manager must notify the change initiator, the change owner, and all stakeholders of the approval and scheduling of changes. All of the affected parties should be encouraged to provide feedback about the impending change project.

4. Change Building, Testing and Implementation

The change manager monitors the progress of the change process and provides feedback to the Steering Committee. Based on this information, the Steering Committee and the change manager evaluate and correct problems and adjust schedules.

5. Change Outcome

At the conclusion of the change process, the Steering Committee and the change manager should provide outcome notification to all stakeholders. This should be done even if the implementation was not successful. Notification should be provided as soon as possible following the change implementation process. During the 1- to 2-week change evaluation phase (monitoring the change in the IT environment), stakeholders should be kept up to date about the status of the assessment and any problems that are uncovered.

Notification should also be given to any stakeholders who will be affected by the change review process that the change manager conducts to evaluate the change process.

For failed implementations, proper notification should be provided that details why the implementation failed, the next course of action, and how the stakeholders will be affected

Stakeholders must be informed if it is decided to not implement the change and evaluate other alternatives or cancel the change altogether. Notification of successfully implemented changes will also be communicated to all affected parties. Stakeholders should be informed of any follow-up activities that will be initiated.

6. Post Implementation Evaluation

The final phase of the change management process is post-implementation evaluation. This activity includes evaluating the entire change management process from receipt of CR s through implementation by conducting personnel interviews and reviewing documentation. The main objective of this activity is to assess the effectiveness of the change process. Unsuccessfully implemented changes should also be evaluated so that problems can be identified and corrected before further changes are initiated.

Logging

Upon receipt of an CR , the change manager assigns the request a unique identifier and logs it in the change management log. During the change process, the status of the CR should be updated with status identifiers such as open, closed, approved, rejected, pending, deferred, and so on. The log thus provides a quick reference about the status of all CR s. If a change is submitted as the resolution to a problem record (PR), this number should also be logged in the change management log as a cross reference to track the PR status.

At a minimum, the following information should be maintained in the change management log:

1. The unique identifier
2. The change initiator name, position, and contact information
3. The change owner name, position, and contact information
4. CR open date
5. Description of the CR
6. CR close date
7. CR status
8. PR number if a problem report is issued
9. Description of any problems that occur during the process

Throughout the change process the log should be updated with any problems that occur during the project.

The change log is also utilized to track the status of work orders. The change log should include information such as when the order is issued, the order status, problems, and when the order is closed. The CR schedule is also maintained in the change log and regularly updated to reflect the current status of the change project.

Active Directory Schema Changes

Schema changes require special attention in this document because they are classified as major changes to the forest and will affect all agencies belonging to the forest. Special attention is also warranted because several independent software vendors and in-house development groups may desire to change the schema for use in their applications.

Appendix A contains information about Active Directory schema management and extension best practices. This section outlines the procedures for change management, as it pertains to the schema.

Figure 2 illustrates the schema change process in the change management framework. Most of the same roles exist, but have been updated to reflect the group names currently used.

Schema Change Roles

For the Washington State multi-agency forest deployment, the **Windows 2000 Steering Committee** will serve as the schema Change Board. Collectively, the Windows 2000 Steering Committee will have joint responsibility and joint ownership on the overall change control process.

DIS will coordinate the multi-agency forest deployment, manage the shared root domain, and oversee all enterprise-wide functions, including changes to the schema.

Within DIS, Forest Root Administrators are designated as the "Schema Change Master." It is this person's job to maintain the testing and deployment of the multi-agency schema.

To prevent schema changes by unauthorized personnel, all user accounts, except the one given to the Schema Master, will be deleted from the Schema Administration group in every Active Directory Domain within the shared forest.

Because of the impact the schema has State-wide, all schema changes will be shared with the Microsoft Product Support Account Manager Chris Whitney for review. Microsoft can leverage the knowledge gained of the changes to help in resolving future issues regarding Active Directory within the state agencies.

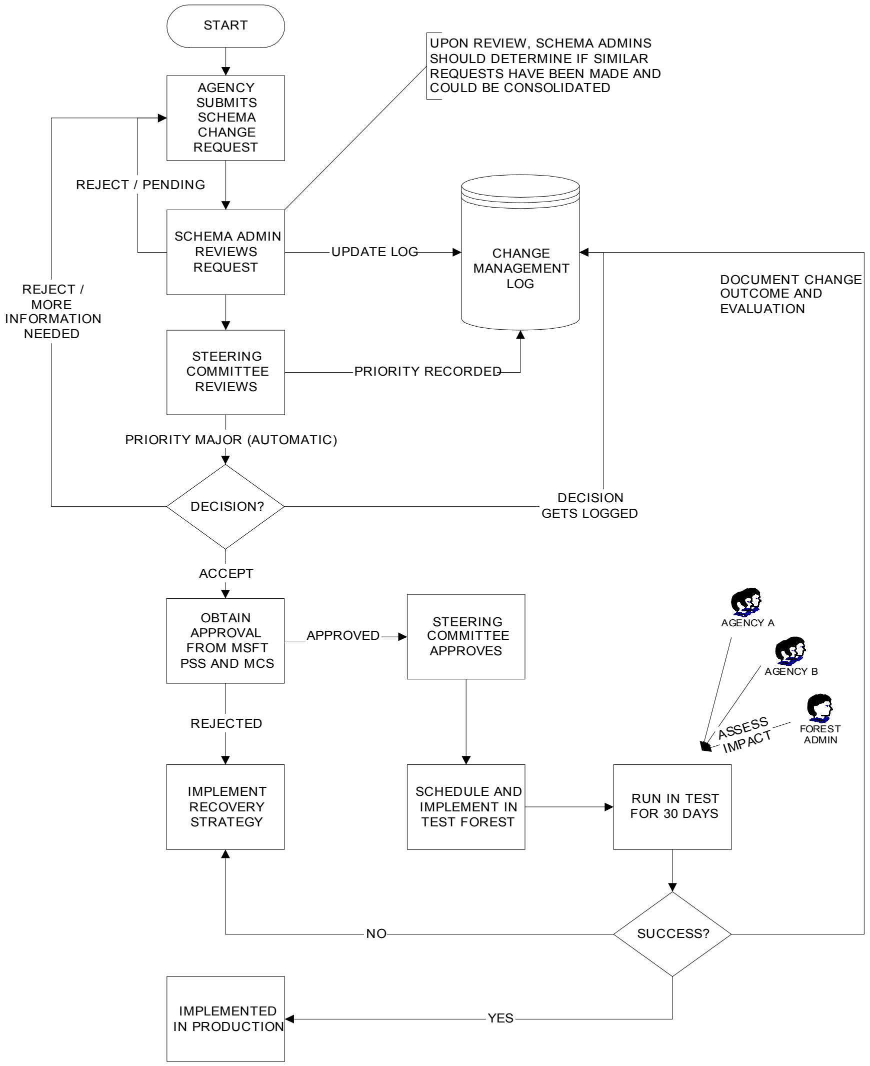
START
AGENCY SUBMITS SCHEMA CHANGE REQUEST
UPON REVIEW, SCHEMA ADMINS SHOULD DETERMINE IF SIMILAR REQUESTS HAVE BEEN MADE AND COULD BE CONSOLIDATED
REJECT / PENDING
SCHEMA ADMIN REVIEWS REQUEST
UPDATE LOG
CHANGE MANAGEMENT LOG
REJECT / MORE INFORMATION NEEDED
DOCUMENT CHANGE OUTCOME AND EVALUATION
STEERING COMMITTEE REVIEWS
PRIORITY RECORDED
PRIORITY MAJOR (AUTOMATIC)
DECISION?
DECISION GETS LOGGED
ACCEPT
OBTAIN APPROVAL FROM MSFT PSS AND MCS
APPROVED
STEERING COMMITTEE APPROVES
AGENCY A
AGENCY B
FOREST ADMIN
ASSESS IMPACT
REJECTED
IMPLEMENT RECOVERY STRATEGY
SCHEDULE AND IMPLEMENT IN TEST FOREST
RUN IN TEST FOR 30 DAYS
NO
SUCCESS?
IMPLEMENTED IN PRODUCTION
YES

Figure 2 - Schema Changes

Schema Change Process

Figure 2 depicts the chains of events for schema change request and approval process:

1. When the need arises, a state agency will first propose a change request to the joint Windows 2000 Steering Committee for review.
2. The Steering Committee verifies that no previous request similar to the current proposal exists. If not, go to step 4.
3. If a previous request exists, then the two requests are compared to determine whether new business justifications are present. If not, the change request is disapproved and closed.
4. The Steering Committee conducts preliminary review of the change request and its accompanying business cases.

Depending on the merit, the process either terminates and the change is disapproved, or the process continues to step 5.

5. The DIS "Schema Master" completes an initial test to evaluate the impact of the changes on existing directory schema and applications. If the changes are deemed feasible, continue. Otherwise, the change request is disapproved and closed.
6. Once Schema has passed initial certification, the proposal is forwarded to the onsite MCS person or Microsoft product support personnel. The person(s) ensure(s) that the proposal will pass supportability review. If the change fails at this point, the request is disapproved and closed. Otherwise, it is forwarded through to the next process.
7. Steering Committee reviews initial test result, and gives the go/no-go for schema change.
8. Changes are implemented in test environment where they are evaluated by all affected personnel and agencies. During this time, evidence is gathered by the change owner that ultimately becomes part of the change outcome and post-change evaluation documentation.
9. At the end of 30 days, the change outcome is determined. If successful, schedule and implement into production. If not, the rollback plan must be implemented.

NOTE: Schema change rollback plans must take into consideration that once a change to the schema is made, it cannot be undone. It can only be disabled.

Special Considerations for Schema Changes

The schema changes for classes and attributes must be described in detail by the change initiator. The description of the change should include the following:

Data Owner	Which agency, person, or group owns the data that this schema change will use (for example, if it is Person data, it might be owned by DOP).
How Used	What is the justification for the change request and what other groups have been identified as potential customers of this data element.
Prefix	If the extension is not using an existing registered prefix it must be registered with an assigned OID and prefix.
OID	Each schema object must be identified with a unique object identifier or (OID). This number uniquely represents the object (attribute or class) and should be unique worldwide. For schema extensions required to support in-house applications, the state agencies should contact an Object Identifier (OID) issuing authority for the

	proper classification. This OID will prevent State of Washington from having duplicate object types declared.
Class Type	Applicable if adding a new schema object class. Consideration also needs to be given to where in the hierarchy the class will be located.

The schema is neither a database nor a file system. Don't treat it as such. It is better to place references in the directory that point to other data stores than to use the directory for something for which it was not designed. The following is a list of guidelines for using the schema:

1. Only define globally interesting, relatively static information in the schema.
2. Objects defined in the schema should not be created very often or modified frequently.
3. Objects should have a long life.
4. Use twice the maximum replication frequency when determining longevity or frequency.
5. Class 88 classes should not be defined in the schema. This will cause problems with ADSI applications.
6. Test your application in a private forest and with other applications before deploying.
7. The schema upgrade must be separate from the application installation.
8. Resist adding attributes to the Global Catalog. This will cause a full GC sync that will affect network performance.

Schema Testing Procedures

Once the schema change request has been approved by the steering committee, it will be implemented in the WA.TST test forest. During this time, the impact of the change, if any, will be documented. As part of the testing process, the following steps will be performed.

1. Modify test forest schema master (FSMO Role) to accept schema extensions. This is done by enabling a write copy of the schema operations master using the MMC utilities in Windows 2000. Procedures are documented in Windows Online Help.
2. Install schema extensions in WA.TST. Extensions may be installed automatically by server software applications (such as Exchange Server) or may be applied by in-house methods.
3. For purposes of production, it is required that all schema changes be automated. This prevents the likelihood of errors as responsibility for the change passes from dev to test to production. The preferred scripting environment is Visual Basic Scripting and ADSI.
4. Test new application with new schema extensions. In the event that the change request was a modification of an existing

schema object, it will be necessary to create a plan for validating the health of every application known to have used that object.

5. Observe the effects on replication, global catalogs and domain trees to determine if hardware resources might be insufficient, bandwidth allocation should be adjusted, or replication schedules changed to handle the new information.

Urgent Change Requests

The change manager may classify CR s as urgent changes. Urgent changes are changes that must be implemented as soon as possible to prevent user downtime and degradation of service that impedes business processes. Changes that are required to restore mission-critical business applications or are required to prevent significant user downtime are examples of urgent changes.

The urgent change process, which allows an organization to restore normal operation as quickly as possible, is an accelerated process that follows the normal change process to the extent that time constraints permit. Keep the number of urgent changes to a minimum because these changes pose a greater risk to the IT environment due to the limited testing that is conducted.

The change initiator also may implement changes that meet the above urgency criteria without prior approval from change management. Either method must involve updating the change log and CMDB and communicating the change implementation to affected stakeholders. Communication will most likely occur during the implementation process or immediately following implementation. Due to time constraints, prior notification typically will not be provided. Users and all other stakeholders must be informed about the change, how it will affect the user functionality and business processes, and what follow-up actions, if any, are required. The change manager and initiator must ensure that change notification is provided as soon as possible before, during, or following the implementation process.

It is extremely important that the change log and CMDB be updated to reflect the implementation of urgent changes. Maintaining the change log allows the Steering Committee and affected parties to gather status and problem information that the change manager and initiator may be unable to provide due to time constraints. The log and CMDB should be updated throughout the change process or, if that is not possible due to time constraints, immediately following the change implementation process.

An CR that is assigned an urgent priority by the change manager is immediately passed to the appropriate organizational groups for change component development and procurement, if required. If time permits, the change is passed to the testing coordinator. In the event the change manager requires additional input before authorizing the change, a meeting of the Steering Committee/executive committee will be called. The change manager may decide to take this approach if unsure about the urgency of the change or if it will require extensive resources to implement. The executive committee will perform the same functions as the Steering Committee (assessing and authorizing changes), but in a shorter timeframe. The change manager and the Steering Committee will re-evaluate unsuccessfully implemented changes and take immediate action to rectify the problem(s) and implement the change properly.

Groups within the organization, such as the service desk, should have the authority to implement certain changes without seeking the approval of change management. This path should only be followed for situations that must be dealt with immediately. The change initiator who in this case is also the change owner will be responsible for formulating rollout and most importantly back-out plans.

Note: Implementation in this manner should occur only if the change initiator is attempting to prevent ***significant user downtime*****.**

The change initiator must also inform all affected parties as soon as possible. When bypassing the change process, it is extremely important that the change initiator notify change management so that the CMDB can be updated to reflect the change. An CR should still be submitted, even if the change has been implemented, to ensure that the appropriate parties are made aware of the change. If not implemented correctly or if problems arise, the change is backed out and submitted to the change manager and Steering Committee for review. The change manager will make the necessary resources available to assist the change initiator in correcting the problem and implementing the change.

Following the successful implementation of urgent changes, the change manager, initiator, and owner (in many cases the latter two will be the same person) should communicate the proper information about the change to all affected stakeholders. The change manager and initiator also will begin the process of post-implementation evaluation and subsequently close the CR if IT leadership and user community leadership sign off on the change. If they do not sign off on the change, it will be backed out and submitted to the Steering Committee for review.

Appendix A – Schema Management Best Practices

Active Directory Schema

The schema contains definitions for the universe of objects that can be stored in a directory, and it enforces the rules that govern both the structure and the content of the directory. In Windows 2000 Active Directory, the schema consists of a set of classes, attributes, and syntaxes that define the objects. A class is a category of objects that share a set of common characteristics. An attribute describes the characteristics of an object. The syntax of objects determines the type of value an attribute can have. The base schema that ships in Windows 2000 contains all of the classes and attributes that are used by Windows 2000 and Windows 2000 components.

To take advantage of the Active Directory, there are requirements from other applications to write there own schema extensions. An example of this would be moving the data that is in the Exchange Directory into the Active Directory. The Active Directory service included with Windows 2000 extends the features of previous windows directory services. One of the powerful new features is the ability to dynamically extend the schema with new attributes and classes and use the new extensions immediately.

Schema Extension Considerations

Schema is encouraged if the normal planning and testing criteria is followed, however, the task should not be taken lightly. The following considerations must be carefully weighed before any schema changes are proposed or implemented:

Irreversible – Any schema modifications, once implemented, cannot be reversed.

Elements can not be deleted – Once a schema element is added, it can only be disabled, never deleted. Even authoritative restore cannot remove items from the schema. Do not add elements you may want to delete at a future date.

Forest wide visibility – Schema changes are global. There is a single schema for an entire forest. The schema is globally replicated, i.e. a copy of the schema exists on every domain controller in the forest. When extensions to the schema are added, collaboration amongst the agencies is required to ensure that attributes and classes are not duplicated. The best way to ensure this is to document all the class and attribute additions.

Replication impact – Changes to the schema must be replicated to every domain controller in the forest.

Global Catalog replication – When deciding whether or not to place an attribute in the global catalog, keep in mind that there is the tradeoff between increased replication and increased disk storage on global catalog servers for (potentially) faster query performance.

Index Considerations – Making an attribute indexed means that directory searches involving that attribute are going to be more efficient than if the attribute had no index. The index is built automatically by a background thread on the directory server. Ideally, indexed attributes should be single valued with highly unique values, evenly distributed across the set of instances. Multi-valued attributes can be indexed, but the cost to build the index is larger in terms of storage, update, and search time. Even with single-valued attributes, keep in mind that the more indexed attributes a class has, the longer it takes to store instances of the class.

Guidelines for Creating Schema Extensions

When NOT to Extend the Schema

The schema should not be extended and objects should not be added to the directory when:

1. The objects needing the schema extensions will be used for less than 2 years.
2. The objects are dynamic by nature (dates, currency rate changes, DHCP addresses, etc).
3. The data needs to be indexed to allow for timely lookups (large data stores which would commonly be housed in a SQL database).
4. The synchronization of the objects is imperative to ensure functionality.
5. The object is confidential but has no security parameters defined to ensure its confidentiality.
6. The object updates need to be tracked or logged.

When to Extend the Schema

Extend the schema only if no existing object class meets your needs. Extending the schema is a relatively complex operation; schema changes are replicated to every Domain Controller in the enterprise. Consider your needs carefully and assess whether an extension is really necessary. If it is, then press on! The extensible schema is one of the most useful and exciting features of the Active Directory.

Types of Schema Extensions

Schema updates should occur in this order:

1. Add new attributes.
2. Add new classes.
3. Add attributes to new or existing classes.

Extending a Class vs. Deriving a Subclass vs. Creating a New Class

There are three ways to extend the schema,

1. Add additional attributes to an existing class and / or add additional possible parents for the class;
2. Derive a new subclass from an existing class - the subclass has all of the attributes of the superclass and any additional attributes you specify ;
3. Create an entirely new class with the attributes you specify.

Extend an Existing Class

It becomes necessary to extend an existing class when:

1. The existing class needs additional attributes but otherwise meets your needs. For example, the Finance Department might want a "purchasing limit" attribute added to the user object for people who are cost center managers and have purchasing authority.
2. There is **no requirement** to identify the class as a separate and distinct subclass.
3. You want to use the existing Directory Manager snap-in to manage the extended attributes of the objects. This requires the addition of property pages to the set defined for the object you are extending.

Derive a Subclass From an Existing Class

It's more appropriate to derive a subclass from an existing class when:

1. The existing class needs additional attributes but otherwise meets your needs.
2. There **is a requirement** to identify the class as a separate and distinct subclass.
3. You want to use the existing Directory Manager snap-in to manage the extended attributes of the objects.
4. An exception is the "user" object. It is usually better to extend the user object directly, rather than sub classing it. Using the objectClass to determine what "kind" of user one is dealing with appears attractive, but you cannot alter the class of an object once it is created. This means that exiting users can never be changed to the new subclass, and instances of the subclass can never be changed back into "standard" users.

Create a New Class

Finally, a new class must be created when no existing class meets your needs.

3rd Party Application Schema Extensions

The most common reason for making schema extensions will be the addition of vendor products. This will include both Microsoft products (i.e. Exchange 2000) and other third party products. The government agencies will have little control over the process by which third party packages choose to modify the schema. Although not being able to control the method of extension, state agencies can provide requirements guideline to evaluate the third party package, and can provide the environment to test third party extensions.

The applications requiring schema extensions should meet the following minimum requirements.

1. The application is required to be enterprise wide.
2. The naming conventions used by the application extension either match the existing approved standards or can be adopted to amend the standards.
3. The attributes populated by the application extension do not conflict with the attributes already in use.
4. The Change Board (Windows 2000 Steering Committee) must approve the extension prior to installing the application.

Initial Requirements for Schema Extensions

In order to make schema changes to the Windows Active Directory, some preparation works are required.

Schema Master – All changes to the schema must be performed on the Schema Master Domain Controller. Only one Domain Controller in the forest plays this role. In Washington State multi-agency forest implementation, the Schema Master Domain Controller role will be held by a Domain Controller in the root domain.

Schema Administrators – The privileges for schema changes are restricted to a group in the forest root domain called *Schema Administrators*. For the local government agencies at the State of Washington, only selected directory administrators at DIS will become members of the Schema Administrators group.

Enable Schema Modification – To make the schema modifiable, follow these procedures:

1. Using the Schema MMC, highlight Active Directory Schema and click Action/Operations Master.
2. Check mark the box "The Schema may be modified on this Domain Controller". The Schema is now ready to be modified.

Tools for Schema Extensions and Administration

When extending the AD schema, four tools are available:

1. Extend using LDIF scripts, which is Microsoft's recommended mechanism.
2. Extend using Comma-Separated Value scripts (CSVDE).
3. Extend programmatically.
4. Extend via the user interface.

The user interface option is definitely the easiest to learn and use, but it has limitations. This option is a great method for the initial creation, but is not self-documenting. In addition, it is easy to make mistakes one time to the next when recreating previous work. LDIF is the best self-documenting tool, as its file format is easy to read. CSVDE's only strong point is that it can represent binary values in Unicode format, but this option fails in most other respects. Using a program, while being self-documenting, is by far the most complex. Therefore, a combination of the above seems most appropriate.

Schema Extension Best Practices

From field knowledge and experiences, Microsoft recommends the following best practices to follow when customizing and extending the Active Directory schema becomes necessary:

1. All schema changes should be planned, tracked and documented carefully.
2. The schema should be modified only when absolutely necessary. Before any modifications to the schema are proposed, one should have a thorough understanding of the default schema that comes with Active Directory. Review the existing schema to verify that there is no way to use the existing classes or attributes for the schema extension that is being requested.
3. Prototype the schema update in a separate test environment. Make schema changes first in a lab environment against a test forest and then apply the extensions to the production forest only after they have gone through a comprehensive testing process.
4. Before any schema extensions take place, first make sure that the schema on each domain controller in the forest is up-to-date. Then, back up the Schema Master server.
5. Be extremely cautious about adding new schema attributes to the Global Catalog. If large number of new attributes in the Active Directory were to replicate to the Global Catalogs, it could saturate the remote Global Catalogs across slow WAN links.
6. Be conservative on the number of types added. Limit the schema addition to as few bytes as possible. If the data is not carefully planned, there is the potential risk of creating a replication nightmare. Many attributes will never be correct

because of the time it takes to replicate those attributes and the network bandwidth to replicate to remote sites.

7. Limit the use of *must-contain* attributes—it may force others across the forest to adhere to the same rules.
8. Manual data entry process can be error prone, so the LDIF export and import method should be used to add extensions to production forest. Using the GUI is recommended only for the initial creation of the extensions in the test forest.

Appendix B – Change Filing Example

As previously noted, any employee in the organization can file an CR . The change manager will need to establish guidelines that determine which employees can submit CR s directly to the change manager and which employees must submit CR s to a reviewing supervisory manager. This layer of review acts as a filter so that the change manager is not inundated with frivolous change requests. The reviewing manager will pass all approved changes to the change manager.

Problem management initiates many CR s. When these CR s are entered into the change management log, the problem number must be annotated so that problem management can cross-reference and track the CR and problem number. Throughout the change process, problem management should be kept up to date on the change status and any problems that arise during the implementation process.

The CR serves as a formal record that is used to track the status of the request from initiation through completion. CR s can be submitted in paper form or electronically. As shown in Figure 4, the CR should provide enough information to allow the change manager/Steering Committee to assess the request, yet be easy to fill out. The change initiator should fill out the first three sections.

The first section of the form provides information about the change initiator and the suggested priority of the change. During the initial assessment of the request, the change manager will confirm the priority level or appropriately adjust it. When initially documenting receipt of the CR , the change manager will record the CR number on the form.

The second section of the form provides a description of the change, the business reason(s) for requesting the change, and the platforms and users that will be affected by the change. The platforms affected include all hardware, software, databases, and so on that will be impacted by the change. In this section the initiator must also describe the backout plans that will be used in the event of a problem with the change implementation. In some cases, this plan will not be known until after the change has been developed and tested. For example, if the change requires the development or procurement of new applications, that backout plan will have to be formulated after the applications have been tested.

The third section of the form provides information about the impacts associated with implementing the change and the training that will be required. For large changes, the Steering Committee will typically assess the impacts based on the CR impact analysis and additional information provided from sources such as the configuration management baseline report.

Change Request Form

Submission Date:	CR Number:	Priority: □ High □ Significant □ Standard □ Low
Initiator Name:	Schema? □ YES □ NO	
Phone:	Urgent? □ YES □ NO	
Department:	Impact On Users: □ All Users □ <1000 □ <100 □ Agency Only	

Description of Change:
Business Reason:
Platforms Affected:
Backout / Recovery Plan Description:
Impact Assessment Results & Training Requirements:

Change Owner:	Business Area:
Home/Pager Number:	Backup Support Personnel:
On-Call Support Personnel:	Home/Pager Number:
Approved By:	Completed By:
Date:	Date:
Planned Implementation Date:	

Implementation Results:

The fourth section contains information about the change owner and the support and backup personnel. The change manager/Steering Committee may decide to appoint a different change owner than the owner provided by the change initiator. The names of support personnel, along with their contact information, must be provided to ensure that the necessary resources are available if problems arise.

The change manager documents the final section. The implementation date is the scheduled date provided by the change manager/Steering Committee. Following the implementation, the change manager records the results. For failed implementations, detailed information should be provided about why the change failed and the next steps that should be taken.

The change initiator should provide as much information as possible. The service desk will provide any assistance required to fill out the form. The change manager may have to obtain additional information after receiving the CR . For instance, it is unlikely that the change initiator and service desk will know exactly how the CR will impact the IT environment or what resources will be required to successfully implement the change.